The Tradeoffs of Societal Computing

Swapneel Sheth   Gail Kaiser

Department of Computer Science, Columbia University, New York, NY 10027

{swapneel, kaiser}@cs.columbia.edu

Abstract

As Social Computing has increasingly captivated the general public, it has become a popular research area for computer scientists. Social Computing research focuses on online social behavior and using artifacts derived from it for providing recommendations and other useful community knowledge. Unfortunately, some of that behavior and knowledge incur societal costs, particularly with regards to Privacy, which is viewed quite differently by different populations as well as regulated differently in different locales. But clever technical solutions to those challenges may impose additional societal costs, e.g., by consuming substantial resources at odds with Green Computing, another major area of societal concern. We propose a new crosscutting research area, *Societal Computing*, that focuses on the technical tradeoffs among computational models and application domains that raise significant societal issues. We highlight some of the relevant research topics and open problems that we foresee in Societal Computing where software engineering research approaches and techniques seem particularly likely to be fruitful. We feel that these topics, and Societal Computing in general, need to gain prominence as they will provide useful avenues of research leading to increasing benefits for society as a whole.

Categories and Subject Descriptors K.4.0 [*Computing Milieux*]: Computers and Society—General

General Terms Human Factors

Keywords Correlation Privacy, Gullibility Factor, Regulatory Localization, Green Computing, Social Networking, Web 2.0

1. Introduction

Today's college students do not remember when social recommendations, such as those provided by Amazon, Netflix, Last.fm, and StumbleUpon, were not commonplace. The rise of Web 2.0 and social networking has popularized social computing as a research area. Established research communities such as Human Factors [44], Computer Supported Cooperative Work, and Software Engineering have fostered emerging topics such as Recommender Systems [17, 23, 47] and Social Software Engineering [8, 18, 27, 38, 39]. However, social computing is primarily concerned with achieving individual benefits from community participation, and not so much with addressing the societal downsides.

We present a novel problem – or perhaps a novel way of looking at known problems – that we believe has not yet been explored by the community. Thus we propose and define "**Societal Computing**," a new research area for computer scientists in general and software engineering researchers in particular, concerned with the impact of computational tradeoffs on societal issues. Societal Computing research will focus on aspects of computer science and software engineering that address significant issues and concerns facing the society as a whole such as Privacy, Climate Change, Green Computing, Sustainability, and Locale-Aware Legal Challenges. In particular, Societal Computing research will focus on the research challenges that arise due to the tradeoffs among the different areas mentioned above. An example of such a tradeoff could be a complex software system that needs to comply with varying laws in different regions or countries. While complying with such laws is important for the society as it might safeguard the interests of individuals, doing so might require investing considerable computer resources through the entire software lifecycle, which might not be a good idea when Green Computing is concerned.

These tradeoffs can affect the entire software lifecycle, from conceptualization and development to deployment and operation. Many Societal Computing issues stem from recent trends in social computing, and possibly may even be solved by drawing on social computing models, such as the wisdom of crowds, collaborative filtering and so on; but many of the concerns are orthogonal and could possibly be addressed by novel approaches grounded in software engineering.

We describe our motivation in the next section and briefly outline some initial Societal Computing topics in the follow-

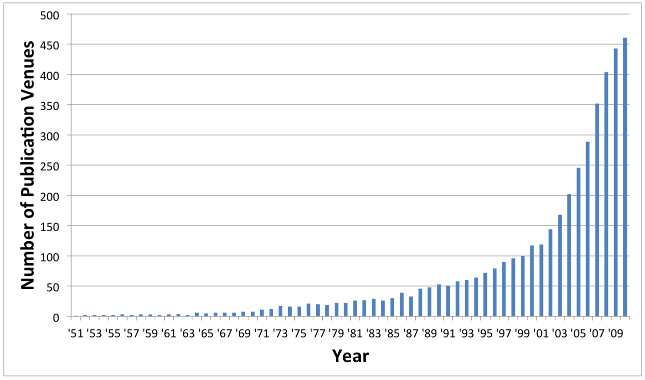

Figure 1. Number of Publication Venues in the ACM Digital Library from 1951 to 2010

ing sections. In this essay, we focus, in particular, on the tradeoffs of these Societal Computing topics and we highlight a few research challenges posed by tensions between prospective technologies targeting these subareas.

2. Motivation

Anthony Kronman in his book "Education's End: Why Our Colleges and Universities Have Given Up on the Meaning of Life" feels that graduate programs at universities have become increasingly specialized. He argues that initially universities were much more broader in their scope and increasing emphasis on the research ideal has resulted in them becoming very specialized. He says: "Graduate students learn to restrict their attention to a single segment of human knowledge and to accept their incompetence to assess, or even understand, the work of specialists in other areas. [. . .] They are taught to understand that only by accepting the limits of specialization can they ever hope to make an "original contribution" to the ever-growing body of scholarship in which the fruits on research are contained." [21]

In the field of Computer Science as well, this increasing specialization is evident by the increasing number of publication venues that exist now and by how this number has changed over the years. A good indicator of the number of publication venues is the number of proceedings (for conferences and workshops) that are available in the ACM Digital Library [1]. This is shown in Figure 1. We see an exponential increase in the number of the publication venues in the last ten years. As the number of publication venues has increased, it has resulted in specialization of Computer Science into subareas and sub-subareas.

While this research specialization is important and has resulted in our increased understanding of the field, it has also made our scopes very narrow. Our problem is an inverse to that of being a "jack of all trades and a master of none." Researchers have become experts in their specialized subareas (and sub-subareas) on Computer Science while being relatively unaware of the other subareas. Due to this narrowing of scope, researchers are not very aware of the advances made in the other subareas and in particular, the tradeoffs that might exist between them. Advanced research and progress made in one research subarea may have a negative effect on some other research subarea.

An example of a well-known tradeoff in Computer Science is that of abstraction vs. efficiency. Higher levels of abstraction are usually beneficial to programmers as it is easier to think at the level of, say, arrays, rather than worrying about zeros and ones. However, this useful abstraction has a tradeoff that affects the runtime efficiency of the code. In an ideal world, it would be wonderful to have both abstraction and efficiency; but this is not always feasible. This notion of tradeoffs is analogous to the concept of Pareto Efficiency [33] in Economics, which deals with the distribution of goods among a set of individuals in society. Pareto Efficiency refers to the state of distribution where it's not possible to make an individual "better off" without making some other individual "worse off." Not being in a Pareto Efficient state would imply that it is possible to optimize both (or multiple) areas; being in a Pareto Efficient state would imply that it is not possible to optimize one area without affecting the other one. In our case of Societal Computing, identifying such a state will be an important research challenge and this identification may not be possible without a detailed understanding of the different areas that we're trying to optimize.

We feel that such tradeoffs exist in many different areas and that a broadening of research scope is necessary to effectively address them. We need to take a much more holistic view of research. We describe some subareas of Societal Computing and the tradeoffs among them in the following sections.

3. Societal Computing Topics

In this section, we describe some research areas relevant to Societal Computing and we will highlight the tradeoffs among these areas in Section 4.

3.1 Privacy

Privacy in the context of social computing systems has become a major concern for the society at large. A search for the pair of terms "facebook" and "privacy" gives nearly two billion hits on popular search engines. Recent feature enhancements and policy changes in social networking and recommender applications – as well as their increasingly common use – have exacerbated this issue [10, 15, 19, 48]. With many online systems that range from providing purchasing recommendations to suggesting plausible friends, as well as media attention (e.g., the AOL anonymity-breaking incident reported by the New York Times [4]), both users and non-users of the systems (e.g., friends, family, co-workers, etc. mentioned or photographed by users) are growing more and more concerned about their personal privacy [40].

Social computing systems, when treated in combination, have created a threat that we call "Correlation Privacy." Narayanan and Shmatikov [29] demonstrated a relatively straightforward method to breach privacy and identify individuals by correlating anonymized Netflix movie rating data with public IMDb data. A similar approach could potentially be applied to any combination of such data-gathering systems, so how to safeguard again these "attacks" may be a fruitful research direction. This is analogous to earlier work addressing queries on census data but, at that time, relatively few prospective attackers [2, 7]. There has been some initial work towards retaining privacy while still benefiting from recommendation systems (e.g., [9, 41]). There have also been other approaches such as k-anonymity [43], differential privacy [12], and applications of differential privacy to other domains [34, 37].

A related challenge is to make the existence of privacy threats more understandable to ordinary users who do not have a technical background and/or in cases where it's not very clear how users' information might be employed by the system, particularly germane for systems that provide APIs making it easier (than screen scraping) for third parties to utilize that information (e.g., [3, 14, 22]). There has been some recent work (e.g., [20, 42]) towards this end. One interesting option might be to make privacy more quantifiable, perhaps by introducing a notion of "Gullibility Factor" for privacy settings, say ranging from 0 to 1 with 1 being the least private. For example, we could say that the default settings for Facebook have a Gullibility Factor of 0.5, whereas for Twitter it's 0.2 (we made up these numbers). A simple scoring scheme might steer away fearful users, while encouraging the merely puzzled to consult one of the numerous "how to" guide articles on privacy settings from sources such as the New York Times [36] and the BBC [35].

Software engineering researchers might contribute to addressing these societal problems by investigating: abstractions and programming paradigms for privacy-retaining APIs; human and machine readable notations for expressing how the use of alternative API methods and options affect privacy, with tools to simplify detection of mismatches when integrating components; and testing techniques for detecting privacy weaknesses and vulnerabilities that can work with third-party applications and mashups (e.g., without specifications or source code access).

We have described how Privacy research will play a crucial role in Societal Computing. While research efforts in this area have been promising, there is a lot of scope for further research. Researchers will need to be aware, in particular, of the tradeoff of Privacy with other Societal Computing topics and we elaborate on this in Section 4.

3.2 Locale-Aware Legal Challenges

We now describe another area of Societal Computing that, we feel, will become increasingly important.

Legal issues present additional research challenges for a wide range of software systems beyond just social computing applications. For example, privacy-related laws vary tremendously from country to country, e.g., consider Germany compared to the United States. Systems such as Facebook and Google Street View, which have been accepted by the government and individuals in the US, are facing many obstacles in Germany [40]. Germany has very strict privacy laws after the Second World War to prevent the government from persecuting its citizens. It is illegal in Germany to publish names or images of private individuals (including felons) without their permission [30]. Before allowing Google's Street View service, German data protection agencies asked Google to audit the information collected by their street view mapping cars. During the audit, they discovered that the cars were collecting personal information such as emails and phone numbers from unsecured wifi networks [5].

One intriguing example open research problem would consider the implications of regulation diversity on software engineering. How could we architect complex software systems so that it is easy to build families of systems whose members appropriately configure themselves to the locality of deployment? What design patterns would enable cloud-computing systems self-configure to the locality and/or role of each end-user? To deal with different laws in the different countries, we would require novel modularization mechanisms beyond those employed for software localizations of keyboard, written language, the customs of different geographical regions, etc. We call this "Regulatory Localization." In addition to software engineering, we feel that multidisciplinary research with other areas of Computer Science such as natural language processing would be highly beneficial.

A recent paper [16] also discusses how local laws can affect software engineering. Those authors focus on intellectual property laws and licensing, warranty and liability, and transborder data flows, and propose "lawful software engineering" research directions concerned with coping with the wide variety of legal constraints during software development and deployment. While that work falls within the scope of our proposal, we are primarily concerned with the potential interactions with other aspects of Societal Computing.

3.3 Green Computing

Green Computing (or Green IT) is "the study and practice of designing, manufacturing, using, and disposing of computers, servers, and associated subsystems [...] efficiently and effectively with minimal or no impact on the environment" [28]. With our oil reserves projected to exhaust in less than fifty years [45], and renewable energy sources still providing only a small fraction [13], Green Computing here and now is becoming more and more important and, indeed, vital to our unborn future descendants.

Investigating how to build greener software systems from a software engineering perspective, in addition to the complementary algorithmic efficiency and systems perspective such as resource allocation, platform virtualization, and power management pursued by other computer science subdisciplines [25] will be important. For instance, say we could quantify complex software systems' behavior in terms of energy expended. There has been some initial research in this area such as [46], which tries to quantify the carbon footprint of a Google Search. Then the software engineering community could investigate ways to make this quantification more modular, devise software architectures and design patterns intended to give developers and end-users more control over energy use, and invent testing methods that check for energy violations. And, further, rethink testing in general, perhaps pushing more testing into the field ("perpetual testing" [32]), to reduce pre-deployment energy consumption and, perhaps, better spread the burden across energy sources. If we could make this quantification more modular (perhaps to the level of individual functionality provided by large systems), we could then provide easy means for operators and end-users to disable unneeded modules, which may play a critical role in Green Computing, to reduce energy consumption on server farms, desktops, and the increasingly abundant mobile platforms. As a simple example, a system like Netflix could inform each user that it would save X amount of energy to disable automated recommendations and only enable them when and if really needed (note that user altruism is a very different kind of model than charging extra for certain functionality [31]). But quantifying which user-visible functionality saves how much energy may not be easy, particularly when systems are built by integrating components. Here, as with privacy above, software engineering researchers might devise human and machine readable notations for expressing how the use of specific API features effect energy consumption, and develop models for configuring energy-related options across systems of systems.

Our Societal Computing initiative envisions investigating the tradeoffs of Green Computing with the other areas and we highlight these tradeoffs in the following section.

4. Societal Computing Tradeoffs

While there is a lot of potential for novel research in these individual areas of Societal Computing, in this essay we focus on the tradeoffs between these different areas and the research challenges that arise out of these tensions. A central discussion point of this essay is to consider the problem of how software methodologies and technologies aimed at reducing societal costs in one area can sometimes raise societal costs for another. For example, there may be clever ways to engineer social computing and other applications to protect privacy or enforce regulations that inherently consume vast CPU cycles and other resources, which could be considered "anti-green." We need a holistic view.

4.1 Privacy vs. Green Computing

Say we have developed an awesome new social computing system S whose privacy-preservation properties may be suspect. One possible approach would be to try correlating S with other popular social systems, such as Netflix, IMDb, Facebook, Amazon, etc., to determine whether privacy can indeed be breached and to what degree (e.g., are potentially all users at risk, only those who use a specific other social system, or only a small fraction of the latter with unique information). We might do this prior to public use of S, e.g., using an internal test team and/or informed beta testers (who might invent phony identities). Such an experiment could give us an estimate of the likely privacy breaches, and possibly point towards steps that could be taken to safeguard against them.

However, straightforward mechanisms that poke or datamine for potential breaches would likely require substantial computational resources; while this kind of testing may be a good idea where Correlation Privacy is concerned, it may not be so good for Green Computing. And it also does not address correlation against future social computing systems or unexpected uses of our system. So instead we could wait until S has been populated by the general public and then periodically correlate a sampling, which might require fewer resources and/or better distribute the resource burden, as well as draw on other new social systems as they are launched. But by then any privacy threats could be actual rather than hypothetical, and consequent protective measures too late. What design and testing techniques can we devise to balance privacy with green computing, particularly in a context where subsystems might be developed by different organizations? Broadening of research scope will be important to be able to effectively address these concerns.

4.2 Privacy Laws

As countries are increasingly trying to pass new privacy laws [6, 24] and companies are being taken to court and getting fined for privacy violations [11, 26], legal issues dealing with privacy will become even more complex. We believe that as countries mandate new requirements for privacy, there will be an important tradeoff between these laws and privacy issues - in particular, the Gullibility Factor. Say we have an awesome new system S that has users in different parts of the world. As each country might have (slightly) different privacy laws, our system would need to comply with all the different regulations. Imagine a user Fred who is a US citizen. We would need to comply with US regulations in this case and Fred would have set his privacy settings as needed. Now if Fred decides to travel to another country (say, Germany) for business or a conference, we might also need to comply with the German regulations for privacy. In addition, we might also need to comply with the EU regulations, which may or may not be the same as the German regulations. Having to comply with all these different regulations will only end up making privacy threats and settings harder to understand for users and might also result in less usable systems. Note that such conflicts and confusions needn't arise due to travel to different countries, but might also exist due to the different city, state, and federal rules. What techniques can we use to make privacy and privacy settings more understandable to ordinary users when we need to also comply with complex legal regulations? An understanding of the different research areas involved will be crucial to address the various research challenges that we face.

4.3 Green Computing vs. Green Computing

There is also an interesting (and recursive) tradeoff of Green Computing with itself. As part of the development of greener software systems, we may need to invest substantial computer resources. For example, social recommendation systems tend to rely on expensive data-mining, but developing a greener recommendation system that is kinder to the environment could also be quite expensive. In the worst case, the amount of resources spent on building such green systems may far outweigh the energy benefits of replacing their less-green counterparts with these new systems, a classic example of being "penny wise, pound foolish." How can we efficiently analyze this in advance of expending those resources?

A common theme in these tradeoffs is finding the right balance between the different areas of Societal Computing. If we haven't reached the Pareto Efficient state yet, it might be possible to optimize different areas simultaneously. Once we reach the Pareto Efficient state, trying to improve one of these areas might have an adverse effect of some other area. An important concern will be trying to identify such a state and we believe that this will require a detailed understanding of the various Societal Computing areas. What to do once we reach the Pareto Efficient state gives us further food for thought. One approach to consider, even though it might be considered an anathema to all technological advances, is to spend more *human* time to reduce reliance on non-renewable resources. Most technology (since the dawn of time) has been designed to make humans more productive and to reduce the burden of work for humans. However, as resources start becoming scarce, humans may need to take on more of this burden. In a software engineering context, this might imply a greater reliance on design or code review instead of execution testing. We would then need to figure out how we could do reviews across different systems, e.g., to manually find Correlation Privacy problems. We might also encourage human policing of privacy violations and/or time spent in end-user training to reduce the Gullibility Factor rather than automated ways for detecting these.

5. Conclusion

We have coined a new term: "**Societal Computing**." We feel that the various aspects of Societal Computing should gain increasing prominence in the near future and provide useful and important questions for software engineering researchers whose answers will benefit society as a whole. We feel that a broadening of research scope is very important and necessary to address the research challenges and in particular, the tradeoffs among the different areas of Societal Computing. We have highlighted some of these areas and a few of the research challenges and open problems that we foresee, including the potential for conflicts and tradeoffs among the subareas. Finding the right balance among the tradeoffs in these different research areas will be crucial.

Acknowledgments

The authors are members of the Programming Systems Lab, funded in part by NSF CNS-0905246, CNS-0717544, CNS-0627473 and CNS-0426623, and NIH 2 U54 CA121852-01A1.

References

[1] ACM. ACM Proceedings. `http://dl.acm.org/proceedings.cfm`, 2011.

[2] N. R. Adam and J. C. Worthmann. Security-control methods for statistical databases: a comparative study. *ACM Comput. Surv.*, 21(4):515–556, 1989. ISSN 0360-0300. doi: http://doi.acm.org/10.1145/76894.76895.

[3] Apple Developer. iOS Dev Center. `http://developer.apple.com/devcenter/ios/index.action`, 2007.

[4] M. Barbaro, T. Zeller, and S. Hansell. A face is exposed for AOL searcher no. 4417749. `http://www.nytimes.com/2006/08/09/technology/09aol.html`, August 2006.

[5] BBC. German Street View goes live with enhanced privacy. `http://www.bbc.co.uk/news/technology-11673117`, November 2010.

[6] BBC. Governments 'not ready' for new European privacy law. `http://www.bbc.co.uk/news/technology-12677534`, March 2011.

[7] L. L. Beck. A security mechanism for statistical database. *ACM Trans. Database Syst.*, 5(3):316–3338, 1980. ISSN 0362-5915. doi: http://doi.acm.org/10.1145/320613.320617.

[8] A. Begel, K. Y. Phang, and T. Zimmermann. Codebook: discovering and exploiting relationships in software repositories. In *ICSE '10: Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering*, pages 125–134, New York, NY, USA, 2010. ACM. ISBN 978-1-60558-719-6. doi: http://doi.acm.org/10.1145/1806799.1806821.

[9] S. Berkovsky, Y. Eytani, T. Kuflik, and F. Ricci. Enhancing privacy and preserving accuracy of a distributed collaborative filtering. In *RecSys '07: Proc. of the 2007 ACM conf. on Recommender systems*, pages 9–16, 2007. ISBN 978-1-59593-730–8. doi: http://doi.acm.org/10.1145/1297231.1297234.

[10] B. Bosker. Facebook CEO 'Doesn't Believe In Privacy'. http://www.huffingtonpost.com/2010/04/29/zuckerberg-privacy-stance_n_556679.html, April 2010.

[11] N. Cohen. It's Tracking Your Every Move and You May Not Even Know. http://www.nytimes.com/2011/03/26/business/media/26privacy.html, March 2011.

[12] C. Dwork. Differential privacy. *IN ICALP*, 2:1–12, 2006. doi: 10.1.1.83.7534. URL http://research.microsoft.com/en-us/projects/databaseprivacy/dwork.pdf.

[13] U. Energy Information Administration. International Energy Outlook 2010 - Highlights. http://www.eia.doe.gov/oiaf/ieo/highlights.html, May 2010.

[14] Facebook. Facebook Developers. http://developers.facebook.com/, 2007.

[15] D. Fletcher. How Facebook Is Redefining Privacy. http://www.time.com/time/business/article/0,8599,1990582.html, May 2010.

[16] D. M. German, J. H. Webber, and M. Di Penta. Lawful software engineering. In *Proceedings of the FSE/SDP workshop on Future of software engineering research*, FoSER '10, pages 129–132, New York, NY, USA, 2010. ACM. ISBN 978-1-4503-0427-6. doi: http://doi.acm.org/10.1145/1882362.1882390. URL http://doi.acm.org/10.1145/1882362.1882390.

[17] W. Geyer, C. Dugan, D. R. Millen, M. Muller, and J. Freyne. Recommending topics for self-descriptions in online user profiles. In *RecSys '08: Proc. of the 2008 ACM conference on Recommender systems*, pages 59–66, 2008. ISBN 978-1-60558-093-7. doi: http://doi.acm.org/10.1145/1454008.1454019.

[18] I. Hadar, S. Sherman, and O. Hazzan. Learning Human Aspects of Collaborative Software Development. *Journal of Information Systems Education*, 19(3):311–319, 2008.

[19] S. Johnson. Web Privacy: In Praise of Oversharing. http://www.time.com/time/business/article/0,8599,1990586.html, May 2010.

[20] P. G. Kelley, L. Cesca, J. Bresee, and L. F. Cranor. Standardizing privacy notices: an online study of the nutrition label approach. In *Proceedings of the 28th international conference on Human factors in computing systems*, CHI '10, pages 1573–1582, New York, NY, USA, 2010. ACM. ISBN 978-1-60558-929-9. doi: http://doi.acm.org/10.1145/1753326.1753561. URL http://doi.acm.org/10.1145/1753326.1753561.

[21] A. Kronman. *Education's End: Why Our Colleges and Universities Have Given Up on the Meaning of Life*. Yale University Press, 2007. ISBN 0300122888.

[22] Last.fm. API. http://www.last.fm/api, 2009.

[23] S. L. Lim, D. Quercia, and A. Finkelstein. Stakenet: using social networks to analyse the stakeholders of large-scale software projects. In *Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering - Volume 1*, ICSE '10, pages 295–304, New York, NY, USA, 2010. ACM. ISBN 978-1-60558-719-6. doi: http://doi.acm.org/10.1145/1806799.1806844. URL http://doi.acm.org/10.1145/1806799.1806844.

[24] J. Menn. White House calls for online privacy law. http://www.ft.com/cms/s/0/7267c2c4-500d-11e0-9ad1-00144feab49a.html, March 2011.

[25] Microsoft. *Green Computing*, volume 18. The Architecture Journal, 2008.

[26] C. C. Miller and T. Vega. Google Introduces New Social Tool and Settles Privacy Charge. http://www.nytimes.com/2011/03/31/technology/31ftc.html, March 2011.

[27] C. Murphy, S. Sheth, G. Kaiser, and L. Wilcox. genSpace: Exploring Social Networking Metaphors for Knowledge Sharing and Scientific Collaborative Work. In *1st Intl. Workshop on Social Software Engg. and Applications*, pages 29–36, September 2008.

[28] S. Murugesan. Harnessing green it: Principles and practices. *IT Professional*, 10(1):24 –33, jan.-feb. 2008. ISSN 1520-9202. doi: 10.1109/MITP.2008.10.

[29] A. Narayanan and V. Shmatikov. How to break anonymity of the netflix prize dataset. *CoRR*, abs/cs/0610105, 2006.

[30] K. O'Brien. Technology Butts Up Against Germany's Privacy Laws. http://www.nytimes.com/2010/07/12/technology/12disconnect.html, July 2010.

[31] A. Odlyzko. A modest proposal for preventing internet congestion. Technical report, AT&T Labs - Research, 1997.

[32] L. Osterweil. Perpetually testing software. In *Proc. of the The Ninth International Software Quality Week*, May 1996.

[33] V. Pareto. The new theories of economics. *The Journal of Political Economy*, 5(4):pp. 485–502, 1897. ISSN 00223808. URL http://www.jstor.org/stable/1821012.

[34] J. Reed, A. J. Aviv, D. Wagner, A. Haeberlen, B. C. Pierce, and J. M. Smith. Differential privacy for collaborative security. In *Proceedings of the Third European Workshop on System Security*, EUROSEC '10, pages 1–7, New York, NY, USA, 2010. ACM. ISBN 978-1-4503-0059-9. doi: http://doi.acm.org/10.1145/1752046.1752047. URL http://doi.acm.org/10.1145/1752046.1752047.

[35] L. Rich. A guide to protecting your privacy on Facebook. http://news.bbc.co.uk/2/hi/programmes/click_online/8717750.stm, June 2010.

[36] R. Richmond. Gadgetwise: A Guide to Facebook's New Privacy Settings. http://gadgetwise.blogs.nytimes.com/2010/05/27/5-steps-to-reset-your-facebook-privacy-settings/, May 2010.

[37] I. Roy, S. T. V. Setty, A. Kilzer, V. Shmatikov, and E. Witchel. Airavat: security and privacy for MapReduce. In *Proceedings of the 7th USENIX conference on Networked systems design and implementation*, NSDI'10, pages 20–20, Berkeley, CA, USA, 2010. USENIX Association. URL http://portal.acm.org/citation.cfm?id=1855711.1855731.

[38] A. Sarma, L. Maccherone, P. Wagstrom, and J. Herbsleb. Tesseract: Interactive visual exploration of socio-technical relationships in software development. In *Proceedings of*

the 31st International Conference on Software Engineering, ICSE '09, pages 23–33, Washington, DC, USA, 2009. IEEE Computer Society. ISBN 978-1-4244-3453-4. doi: http://dx.doi.org/10.1109/ICSE.2009.5070505. URL http://dx.doi.org/10.1109/ICSE.2009.5070505.

[39] Z. M. Saul, V. Filkov, P. Devanbu, and C. Bird. Recommending random walks. In *Proceedings of the the 6th joint meeting of the European software engineering conference and the ACM SIGSOFT symposium on The foundations of software engineering*, ESEC-FSE '07, pages 15–24, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-811-4. doi: http://doi.acm.org/10.1145/1287624.1287629. URL http://doi.acm.org/10.1145/1287624.1287629.

[40] M. Shiels. Germany officials launch legal action against Facebook. http://news.bbc.co.uk/2/hi/technology/8798906.stm, July 2010.

[41] R. Shokri, P. Pedarsani, G. Theodorakopoulos, and J.-P. Hubaux. Preserving privacy in collaborative filtering through distributed aggregation of offline profiles. In *RecSys '09: Proceedings of the third ACM conference on Recommender systems*, pages 157–164, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-435-5. doi: http://doi.acm.org/10.1145/1639714.1639741. URL http://portal.acm.org/ft_gateway.cfm?id=1639741&type=pdf&coll=Portal&dl=GUIDE&CFID=58576904&CFTOKEN=24849979.

[42] S. Spiekermann and L. Cranor. Engineering privacy. *Software Engineering, IEEE Transactions on*, 35(1):67 –82, jan.-feb. 2009. ISSN 0098-5589. doi: 10.1109/TSE.2008.88.

[43] L. Sweeney. k-anonymity: a model for protecting privacy. *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, 10(5):557–570, 2002. ISSN 0218-4885. doi: http://dx.doi.org/10.1142/S0218488502001648.

[44] G. Venolia, J. Tang, R. Cervantes, S. Bly, G. Robertson, B. Lee, and K. Inkpen. Embodied social proxy: mediating interpersonal connection in hub-and-satellite teams. In *CHI '10: Proceedings of the 28th international conference on Human factors in computing systems*, pages 1049–1058, New York, NY, USA, 2010. ACM. ISBN 978-1-60558-929-9. doi: http://doi.acm.org/10.1145/1753326.1753482.

[45] J. Vidal. The end of oil is closer than you think. http://www.guardian.co.uk/science/2005/apr/21/oilandpetrol.news, April 2005.

[46] A. Wissner-Gross. How you can help reduce the footprint of the Web. http://www.timesonline.co.uk/tol/news/environment/article5488934.ece, January 2009.

[47] V. Zanardi and L. Capra. Social ranking: uncovering relevant content using tag-based recommender systems. In *RecSys '08: Proc. of the 2008 ACM Conf. on Recommender systems*, pages 51–58, 2008. ISBN 978-1-60558-093-7. doi: http://doi.acm.org/10.1145/1454008.1454018.

[48] M. Zuckerberg. Making Control Simple. http://blog.facebook.com/blog.php?post=391922327130, May 2010.